Ques 1. [**140 marks**] For the dataset build a CNN architecture(use inbuilt PyTorch functions) of the following specifications: The architecture of the convolutional layers are [block1: $[3 \times 3 \times 16] \times 2$], [block2: $[3 \times 3 \times 32] \times 2$], [block3: $[3 \times 3 \times 64] \times 2$] (pooling is an optional layer, use according to your requirement). Perform the following tasks to analyze the CNN architecture:

(1) Implement a CNN architecture with block1 followed by FCs (Fully Connected), block 1, block2 followed by FCs, and block 1,2, 3, followed by FCs layers and a softmax layer. For all the three architectures apply the Tanh or ReLU activation function on all layers.

(2) Implement Dropout and use i) After convolutional layers, ii) Between FC layers

Deliverables:

(1) Visualize 10 random images from each class of the dataset. [**10 marks**]
0 marks if nothing done, 10 marks if all visualizations are available in the report.

(2) Analyze the accuracy and loss while adding block 1, block 2 and block 3 (with mentioned non-linearities) [**45 marks**]
0 marks if nothing done. 15 marks for each set of experiments provided observations are explained clearly.

(3) Analyze the accuracy and loss while changing the dropout probability. (Try atleast three Dropout probabilities e.g. [.2, .5, .8]) [**15 marks**]
0 marks if nothing done. 5 marks for experiment with each dropout provided observations are explained clearly.

(4) Initialize the neural network weights by following: Zero initialization, Random Initialization and He initialization. Which initialization approach is best and why? [**15 marks**]
0 marks if nothing done. 5 marks for experiment with each initialization provided observations are explained clearly.

(5) In the end, report the best accuracy with model architecture and detail analysis of choosing specific hyperparameters and any augmentation or preprocessing if done. [**15 marks**]
0 marks if nothing done. Rest depends upon the reported analysis.

(6) During demo your are given labels of test data(format will be same as training label), your have to evaluate the test accuracy of your best model. Group with maximum accuracy will get full marks [**30 marks**]
0 marks if nothing done. Rest depends upon the obtained accuracy on the test set.

(7) Analyze the results of the best model when all the activation functions are removed. Justify the performance drop. [**10 marks**]
0 marks if nothing done. 10 marks for required justification.

Dataset Description: A h5 file named classification-data.h5py contains 10000×3072 numpy array corresponding to train and test keys. Take *first* 8000 samples for training and remaining 2000 for validation for all your experiments. Each row of the array stores a 32x32 colour image. The first 1024 entries contain the red channel values, the next 1024 the green, and the final 1024 the blue. The image is stored in row-major order, so that the first 32 entries of the array are the red channel values of the first row of the image. labels – The train key contains a list of 10000 numbers in the range $0 - 9$. The number at index i indicates the label of the i^{th} image in the array data.

Ques 2. [**150 marks**] Automated brain tissue segmentation into white matter(WM), gray matter (GM), and cerebro-spinal fluid (CSF) from magnetic resonance images (MRI) is helpful in the diagnosis of neuro-disorders such as epilepsy, Alzheimer's, multiple sclerosis, etc. You have to design a Fully Convolutional Neural Network (FCN) which can specifically deal with MRI data and precisely segment the tissue into the aforementioned classes. The major challenges for acquaintance are to design a suitable training strategy given a limited training sample (three volumes) with high resolution. Hint: You can sample small 2D patches rather that giving full 2D slice for training the FCN.

(1) Download the dataset from here . The provided dataset is for brain tissue segmentation.
(2) Use any FCN (SegNet)to obtain the baseline performance. Use the interpolation to upsample the feature maps in the decoder and max-pooling to downsample the feature maps in the encoder.
(3) Replace the interpolation with transposed-convolution in part (2). How is the performance effected?
(4) Replace the downsampling with strided-convolution in part (3) to achieve the downsampling. How is the performance effected?
(5) Try some skip connections between the encoder convolutional layer to corresponding decoder convolutional layer and analyse the performance.
(6) The dataset is volumetric (3D) but you can not use 3D convolutional layer based architecture. Always work with 2-D slices. **For slicing(2D image) the volume navigate along third dimension.** (i.e. your volume size is $256\times 128 \times$ **256** so you have **256** slices of 256×128.)

Dataset Description: The zip file contains two folders namely training and validation. The training folder contains three brain volumes and validation folder contains one brain volume all of the same size ($256 \times 128 \times 256$) which you can use for training and validation respectively.
Deliverable:

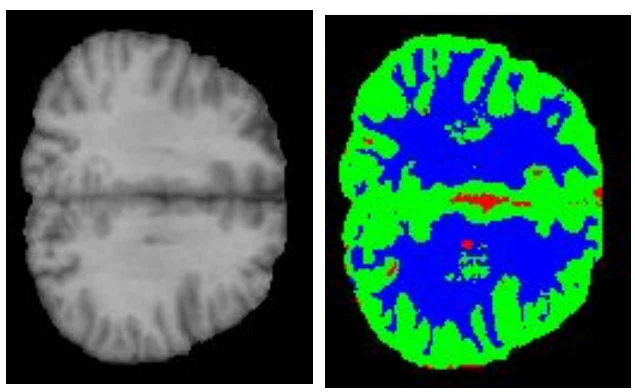

FIGURE 1. Visualization of brain slice. Left: Input slice, right: Segmented slice

(1) Visualize the input and corresponding ground truth for various crosssections (top , middle and bottom area) of the volumes. The ground truth classes are represented by red, green and blue for CSF, GM and WM respectively. For example figure 1 shows the middle slice of a volume and corresponding GT. [**10 marks**]
0 marks if nothing done, 10 marks if all visualizations present
(2) Explanation of your choice of architectures, preprocessing steps, loss functions and training strategy. [**30 marks**]
0 marks if nothing done, 10 marks for preprocessing, 10 marks for architecture, 5 marks for loss functions, 5 marks for training strategy
(3) Analyze the performance (Qualitative-Dice Coefficient/Dice Ratio and confusion matrix, and Quantitative-slice visualization shown in Figure: 1) for (2), (3) and (4) by incorporating the suggested changes with convergence plot. Clearly mention the reason if you improve/not improve DC. [**60 marks**]
0 marks if nothing done, 30 marks for individual implementations of (2), (3) and (4), 15 marks for all convergence plots, 15 marks for qualitative (visualizations) and quantitative (Dice Ratio/DC) results

(4) Try part (5) and report your best model. During demo you are given a separate test data(format same as training dataset) and your have to report the confusion matrix and Dice Ratio. Group with maximum performing model will be given full marks. [**50 marks**]
Marks will be given according to DC value confusion matrix on test set

Ques 3. [**110 marks**] Grab a pre-trained retinanet model for the MS Coco from here. Finetune it on the data given in the same link. Fine-tune the pre-trained model over the eight classes in the given data. Now, with the finetuned model, predict the performance on the vaidation data kept aside. Report the Average Precision (AP) as well as the IoU. Also, show the classwise performance, explain and analyze your findings. Give a detailed explanation of the strategy and preprocessing techniques used for the data, if any. You might need to clean the given data to match the images and annotations.

Deliverables:

(1) Report Average precision (AP) and confusion matrix over the given dataset(validation set) using the pre-trained model(without finetuning). [**15 marks**]
5 marks for reporting AP, 10 marks for confusion matrix; 0 otherwise

(2) Report AP and confusion matrix on the given data using the fine-tuned model. Also, report the classwise performance and analyse your findings. [**15 marks**]
5 marks for reporting AP, 10 marks for confusion matrix; 0 otherwise

(3) Add the loss plots for convergence obtained while training. Justify the choice of the preprocessing and hyperparameters, if any. Also, submit the weights of your best model. [**15 marks**]
10 marks for correct plots, 0 otherwise; 5 marks for preprocessing explanation.

(4) For each class present in the given data, show random examples of images (3 from each class) along with the ground truth bounding box, bbox predictions using the finetuned model as well as the pre-trained model. Compare. [**30 marks**]
15 marks for images on fine-tuned, 15 marks for images on pre-trained. (-2 per missing class)

(5) Write a script to report Average Precision (AP) obtained from the fine-tuned model over the unseen test data (to be released during demo). The format for the test data will be the same as that of the given data, with separate folders for classwise images and an annotation file (*.json* extension). [**35 marks** as per the accuracy]
Marks based on relative accuracy obtained.